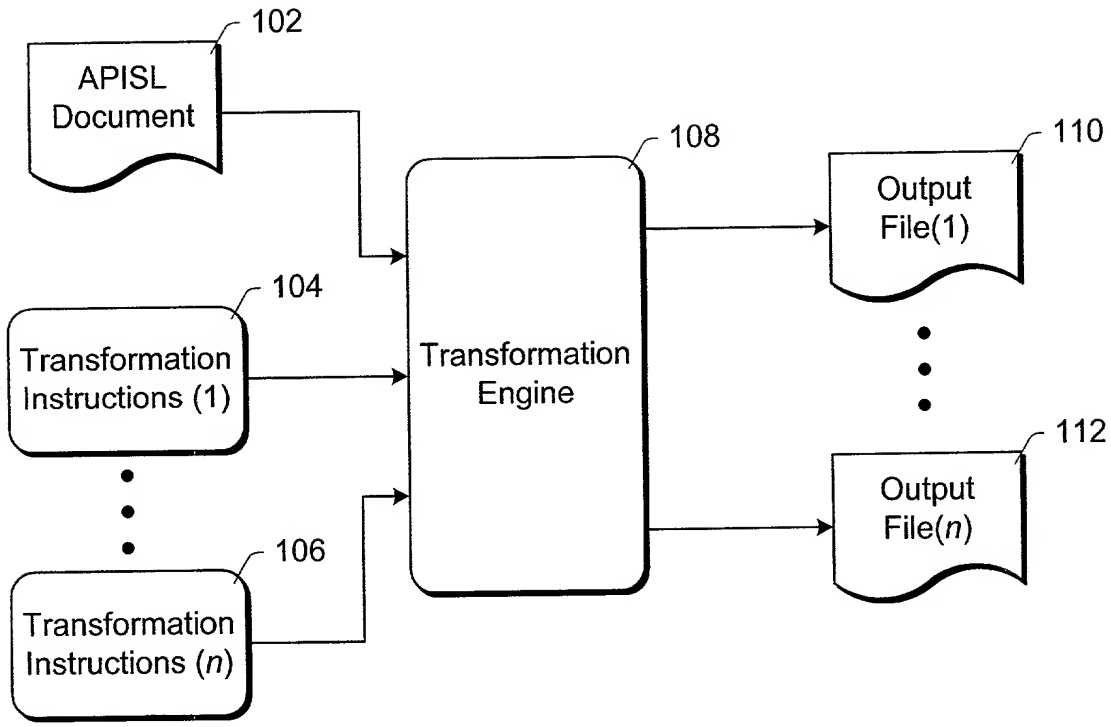

Fig. 1

4005657.02 4005657.02

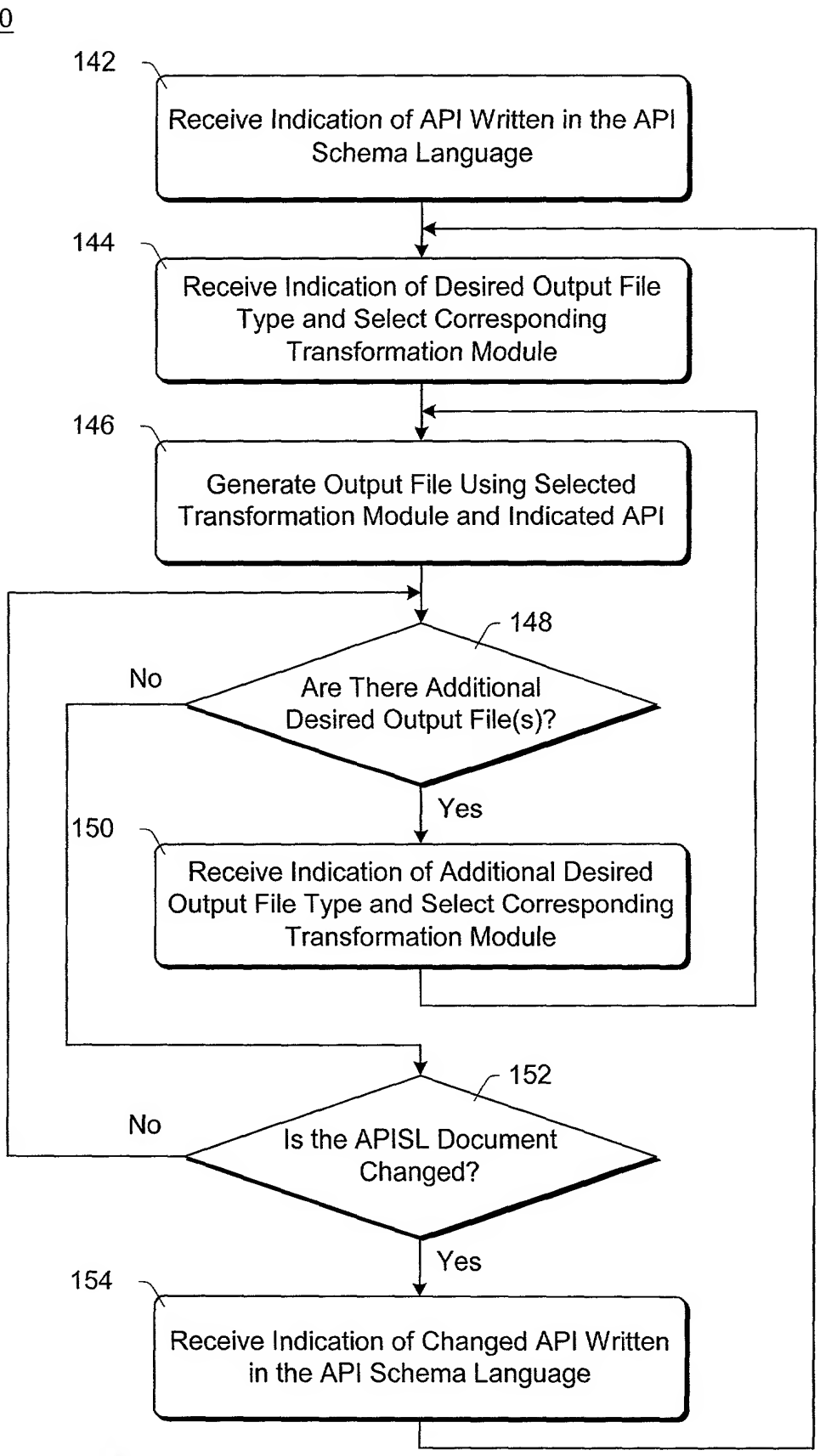
140
142
Receive Indication of API Written in the API Schema Language
144
Receive Indication of Desired Output File Type and Select Corresponding Transformation Module
146
Generate Output File Using Selected Transformation Module and Indicated API
148
Are There Additional Desired Output File(s)?
No
Yes
150
Receive Indication of Additional Desired Output File Type and Select Corresponding Transformation Module
152
Is the APISL Document Changed?
No
Yes
154
Receive Indication of Changed API Written in the API Schema Language

Fig. 2

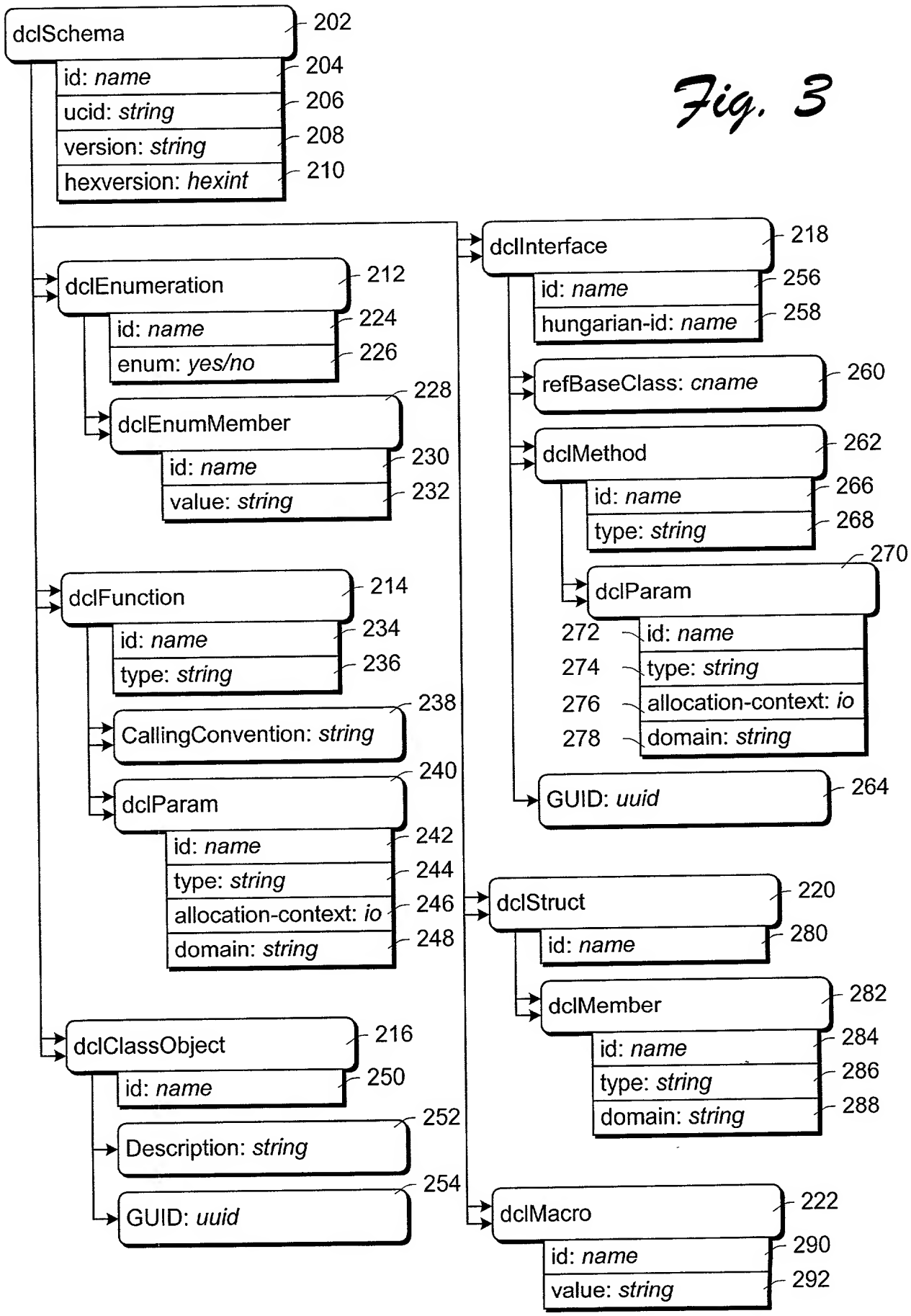

Fig. 3

3054-3057.PDF

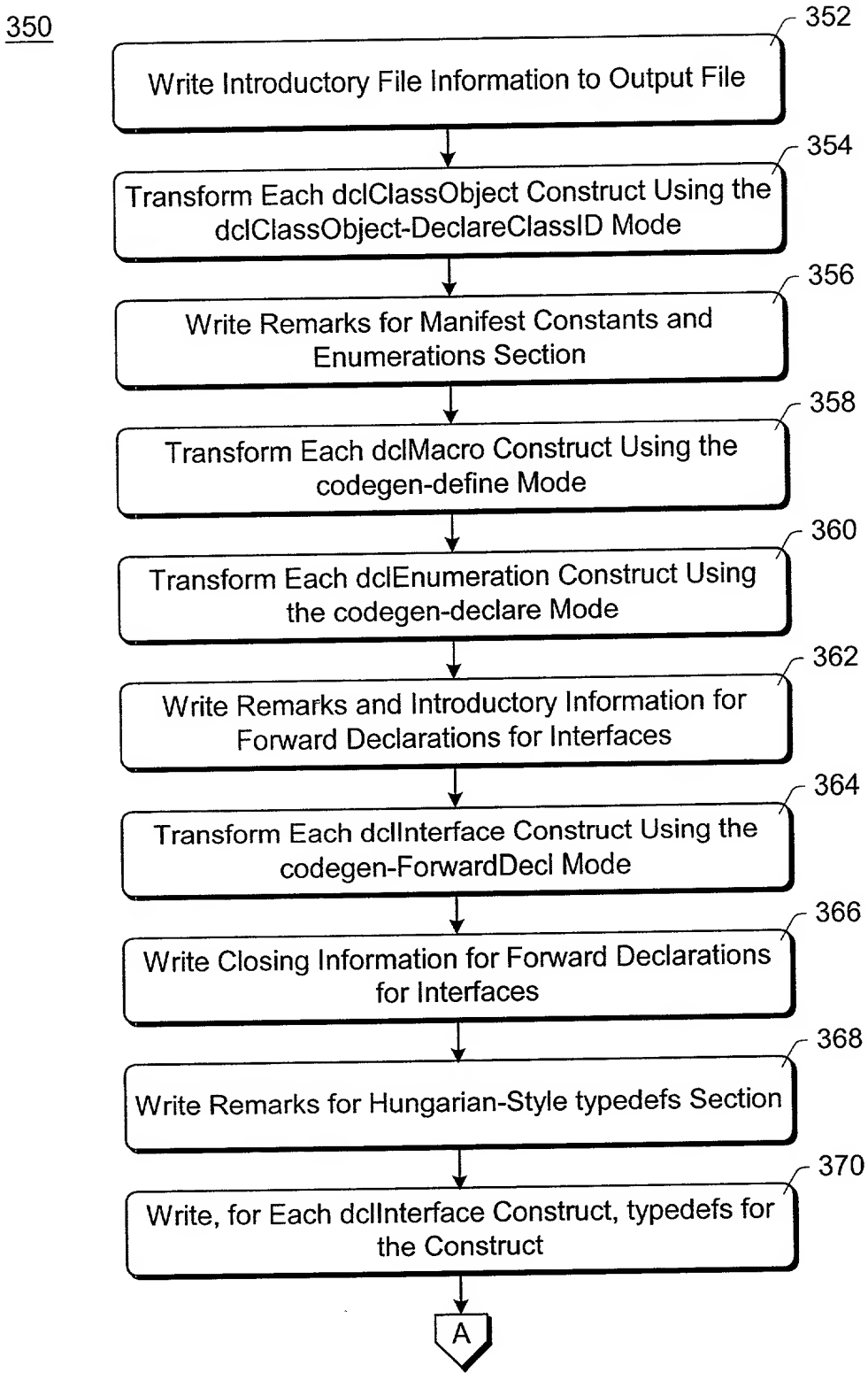
350
352
Write Introductory File Information to Output File
354
Transform Each dclClassObject Construct Using the dclClassObject-DeclareClassID Mode
356
Write Remarks for Manifest Constants and Enumerations Section
358
Transform Each dclMacro Construct Using the codegen-define Mode
360
Transform Each dclEnumeration Construct Using the codegen-declare Mode
362
Write Remarks and Introductory Information for Forward Declarations for Interfaces
364
Transform Each dclInterface Construct Using the codegen-ForwardDecl Mode
366
Write Closing Information for Forward Declarations for Interfaces
368
Write Remarks for Hungarian-Style typedefs Section
370
Write, for Each dclInterface Construct, typedefs for the Construct
A

Fig. 4A

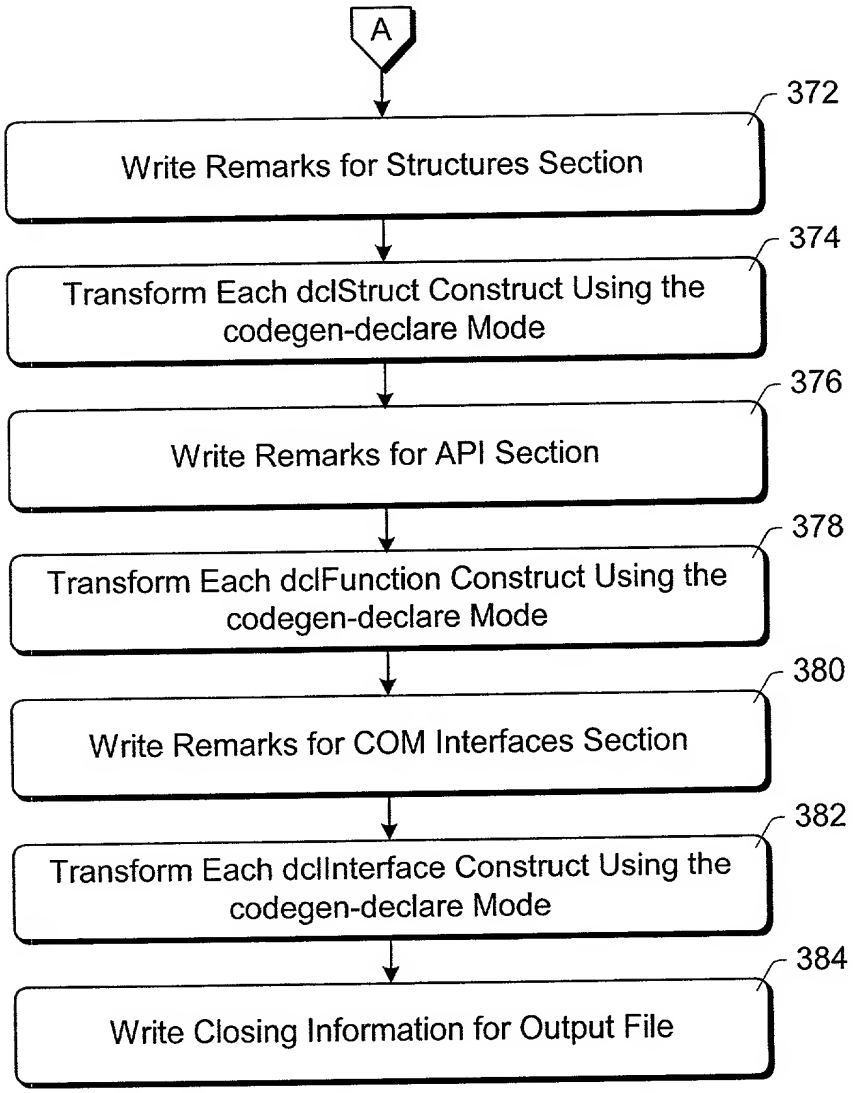

Fig. 4B

Doc7665.doc 4/20/02

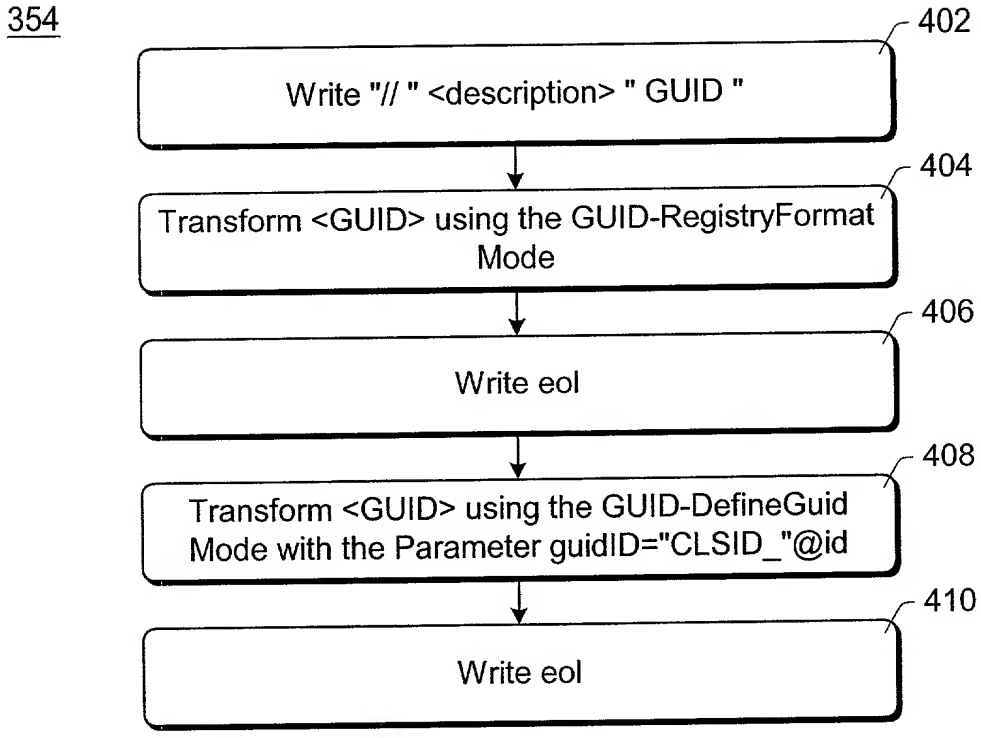

Fig. 5

Doc: 6567.0124

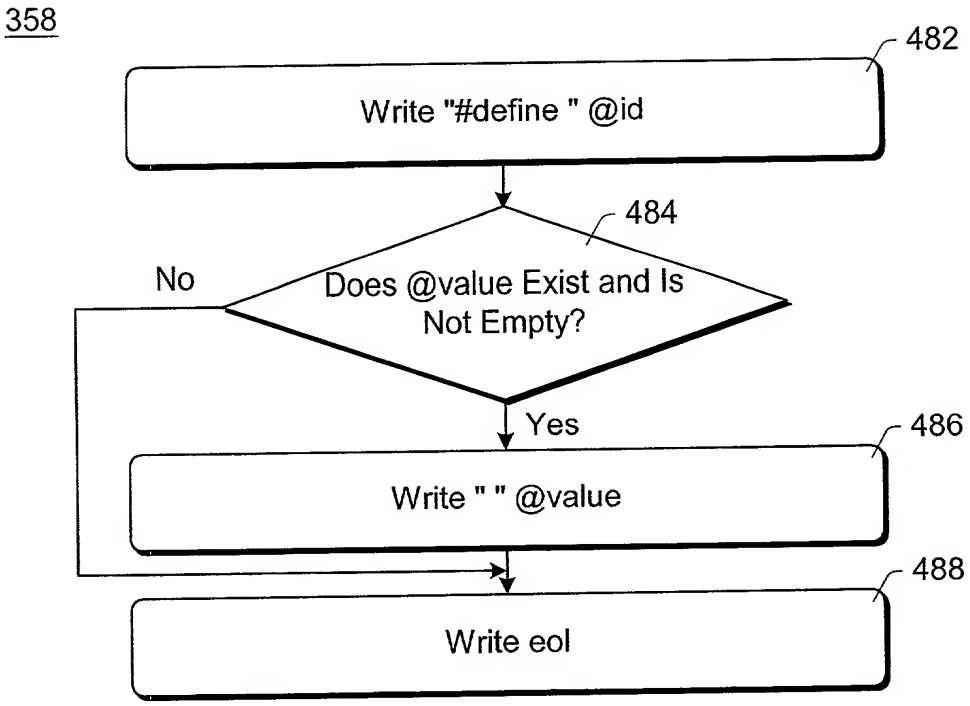

Fig. 6

10976697.doc

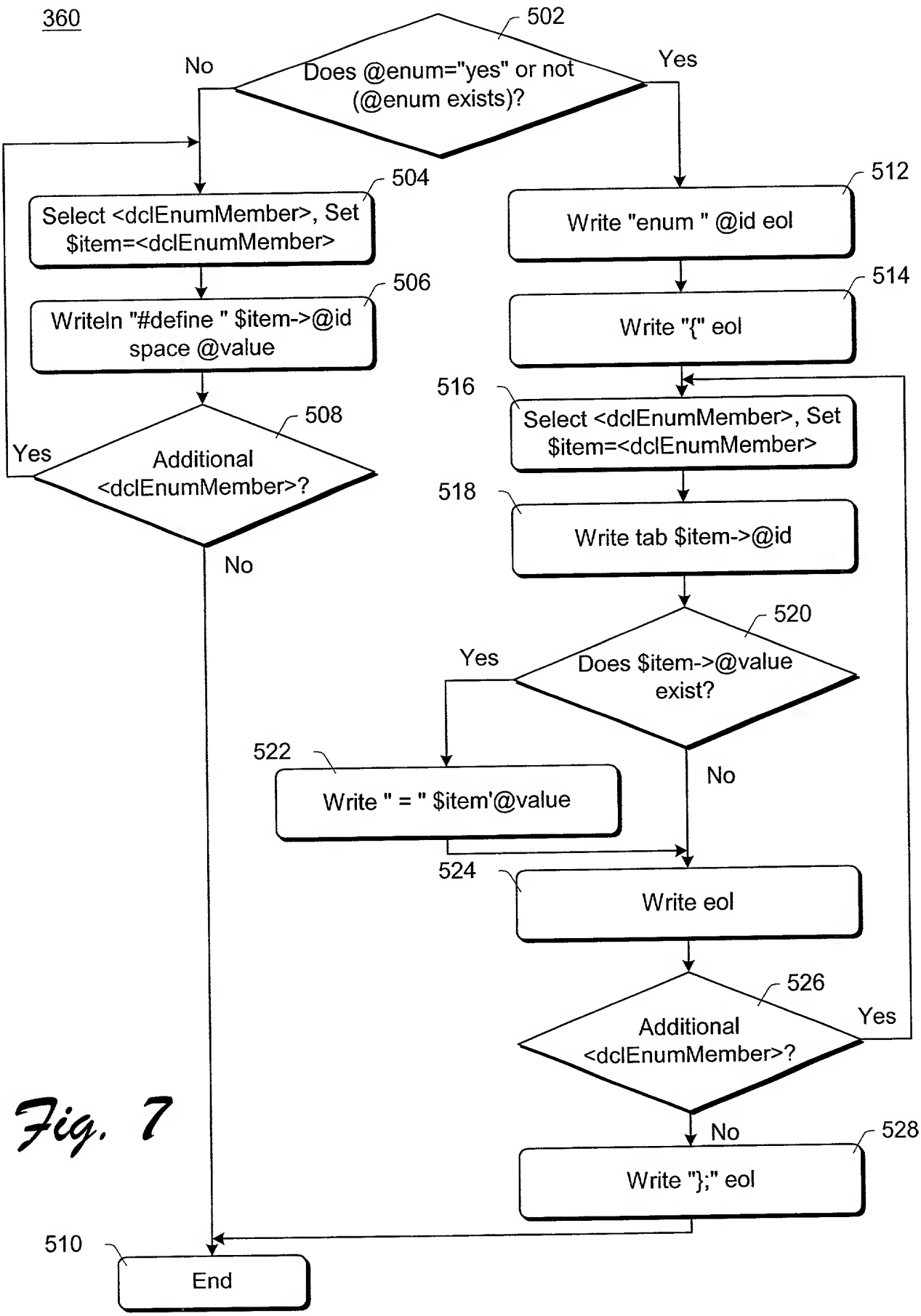

Fig. 7

2004565.02

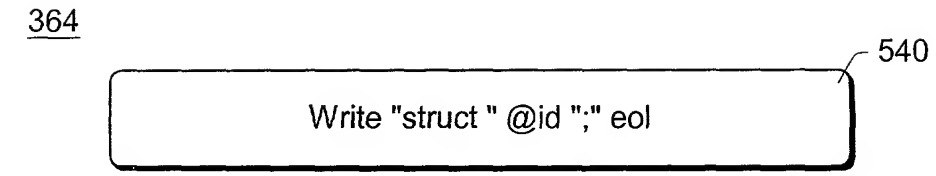

Fig. 8

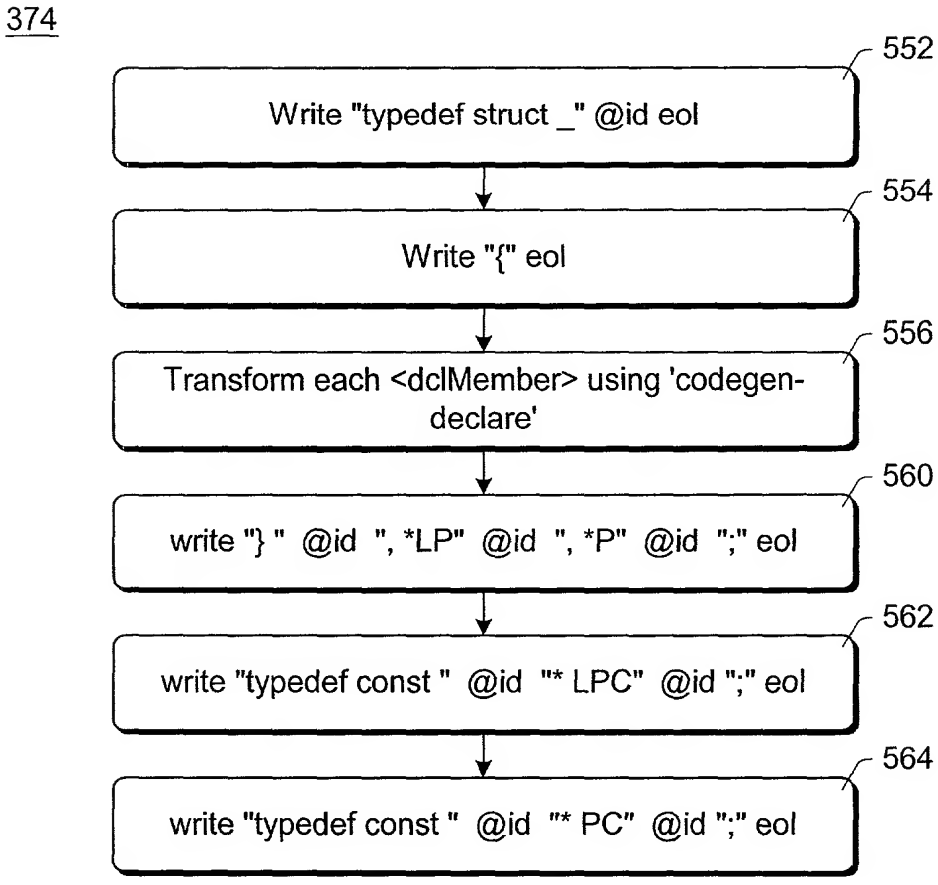

Fig. 9

1000569.0204

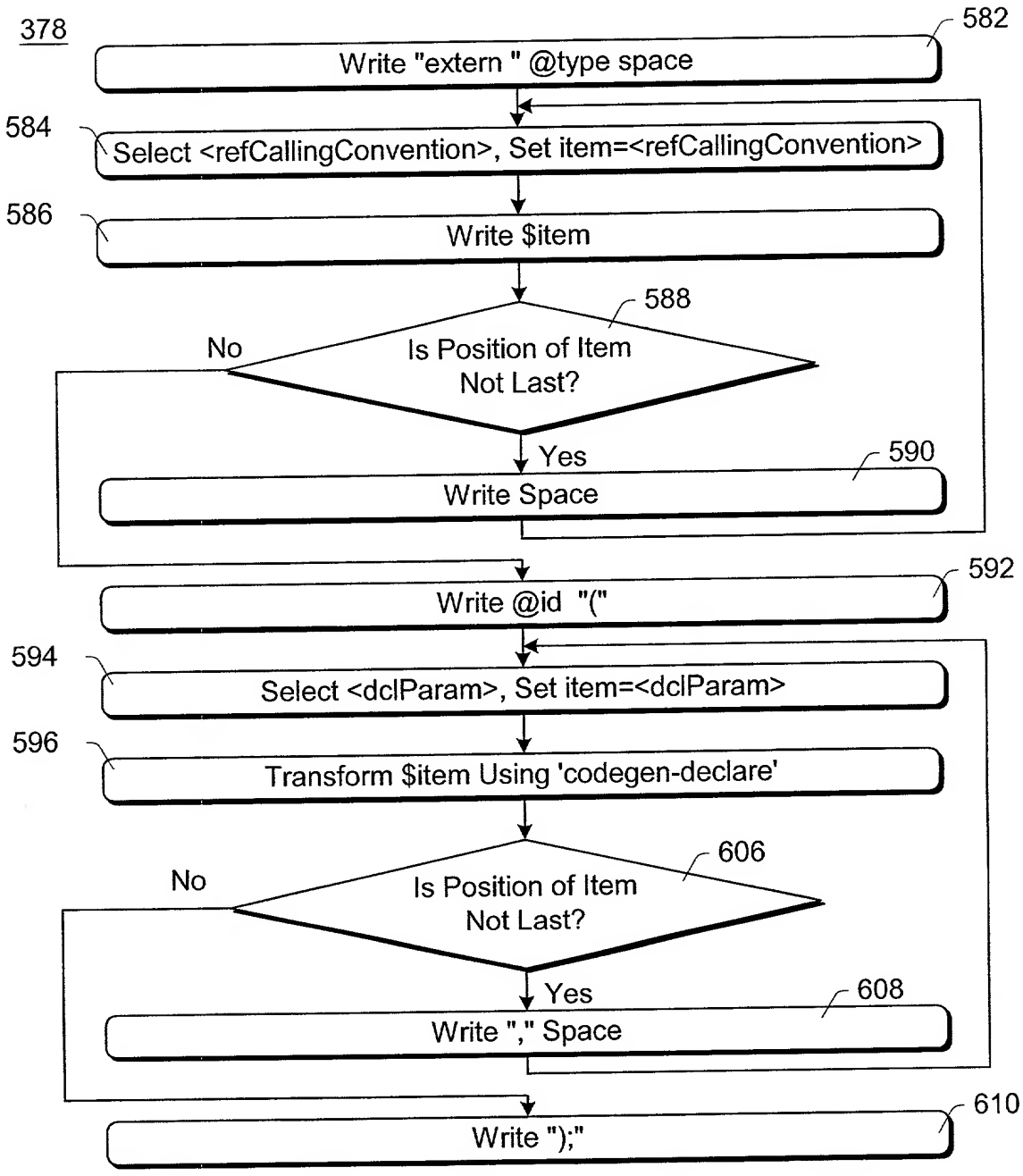
378
582
Write "extern " @type space
584
Select <refCallingConvention>, Set item=<refCallingConvention>
586
Write $item
588
No
Is Position of Item Not Last?
Yes
590
Write Space
592
Write @id "("
594
Select <dclParam>, Set item=<dclParam>
596
Transform $item Using 'codegen-declare'
606
No
Is Position of Item Not Last?
Yes
608
Write "," Space
610
Write ");"

Fig. 10

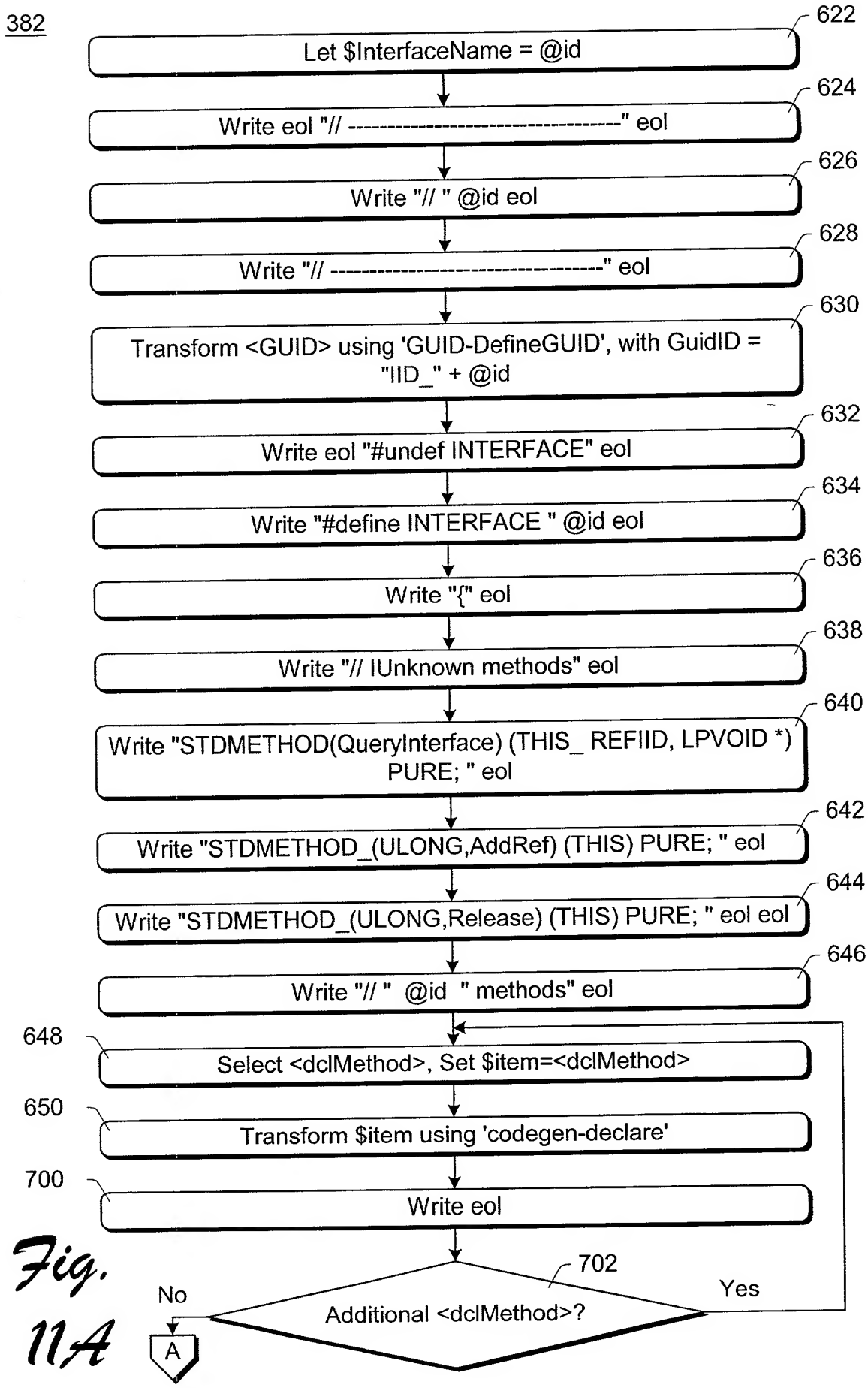

Fig. 11A

4005697.02400

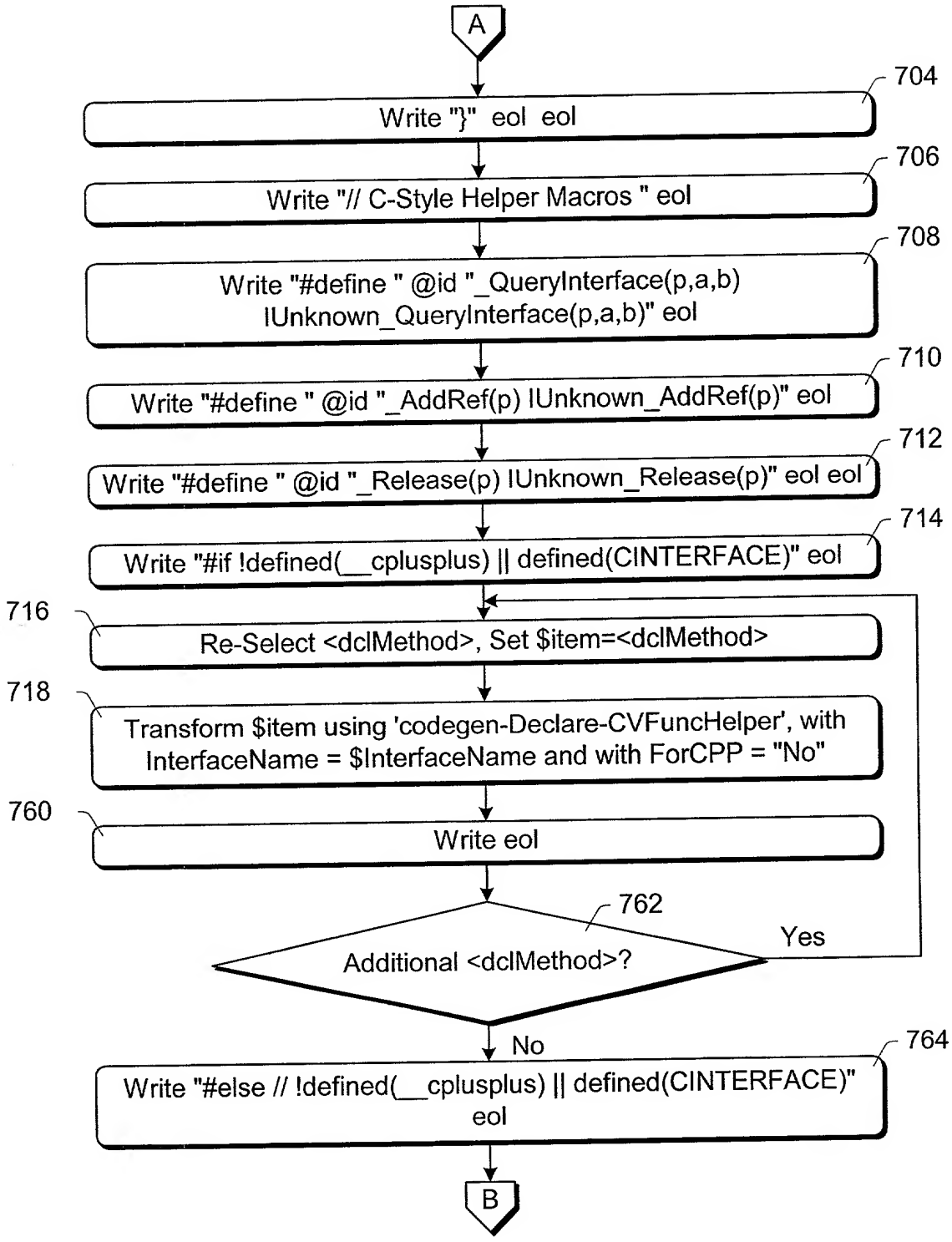
A
704
Write "}" eol eol
706
Write "// C-Style Helper Macros " eol
708
Write "#define " @id "_QueryInterface(p,a,b) IUnknown_QueryInterface(p,a,b)" eol
710
Write "#define " @id "_AddRef(p) IUnknown_AddRef(p)" eol
712
Write "#define " @id "_Release(p) IUnknown_Release(p)" eol eol
714
Write "#if !defined(__cplusplus) || defined(CINTERFACE)" eol
716
Re-Select <dclMethod>, Set $item=<dclMethod>
718
Transform $item using 'codegen-Declare-CVFuncHelper', with InterfaceName = $InterfaceName and with ForCPP = "No"
760
Write eol
762
Additional <dclMethod>?
Yes
No
764
Write "#else // !defined(__cplusplus) || defined(CINTERFACE)" eol
B

Fig. 11B

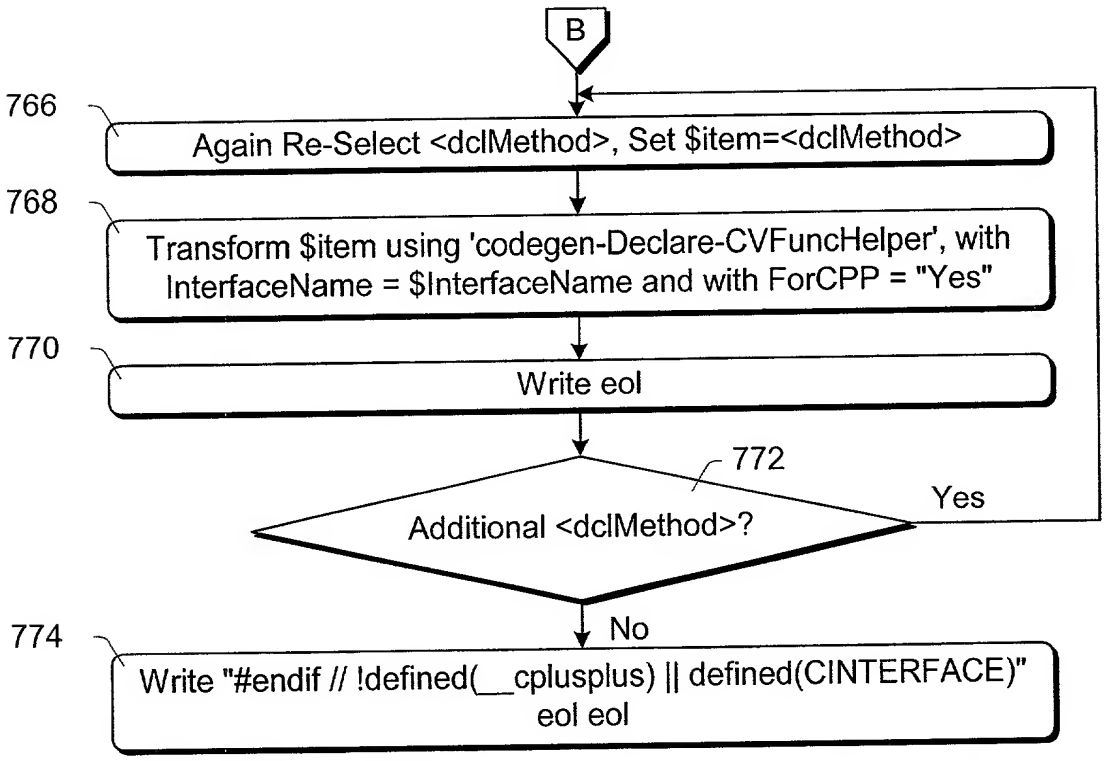

Fig. 11C

Doc-ID: 20040060760

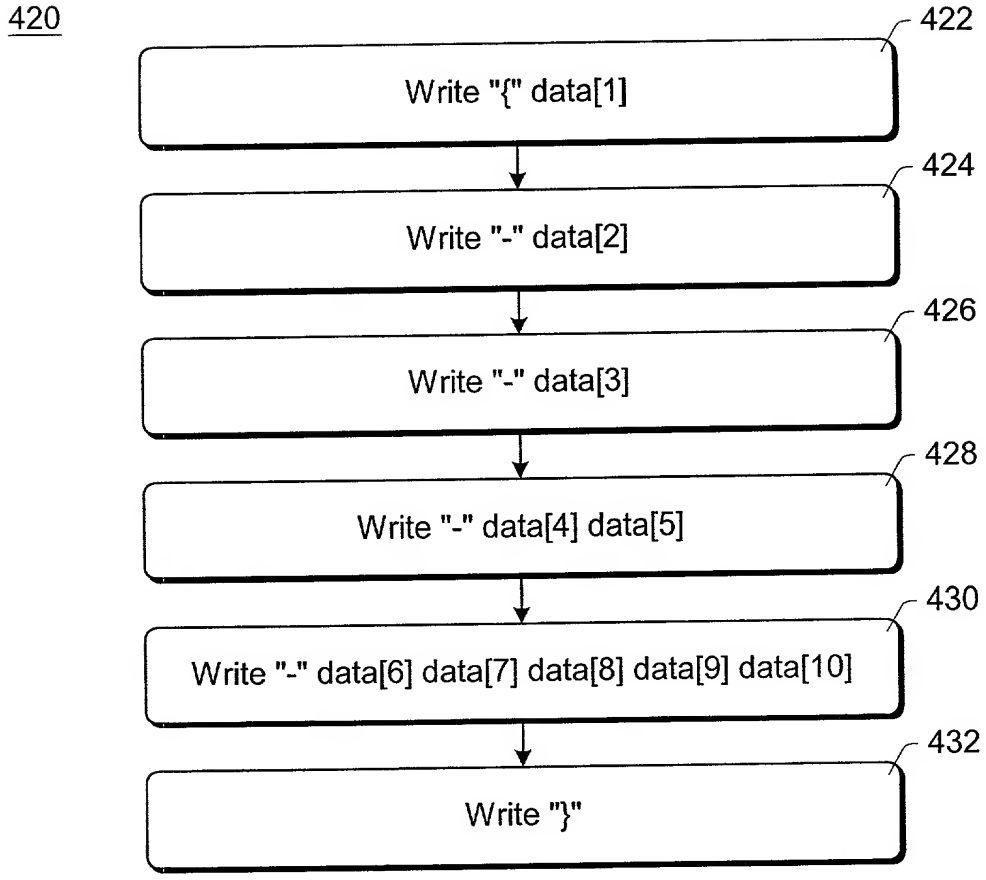

Fig. 12

1007560.02 - 08450

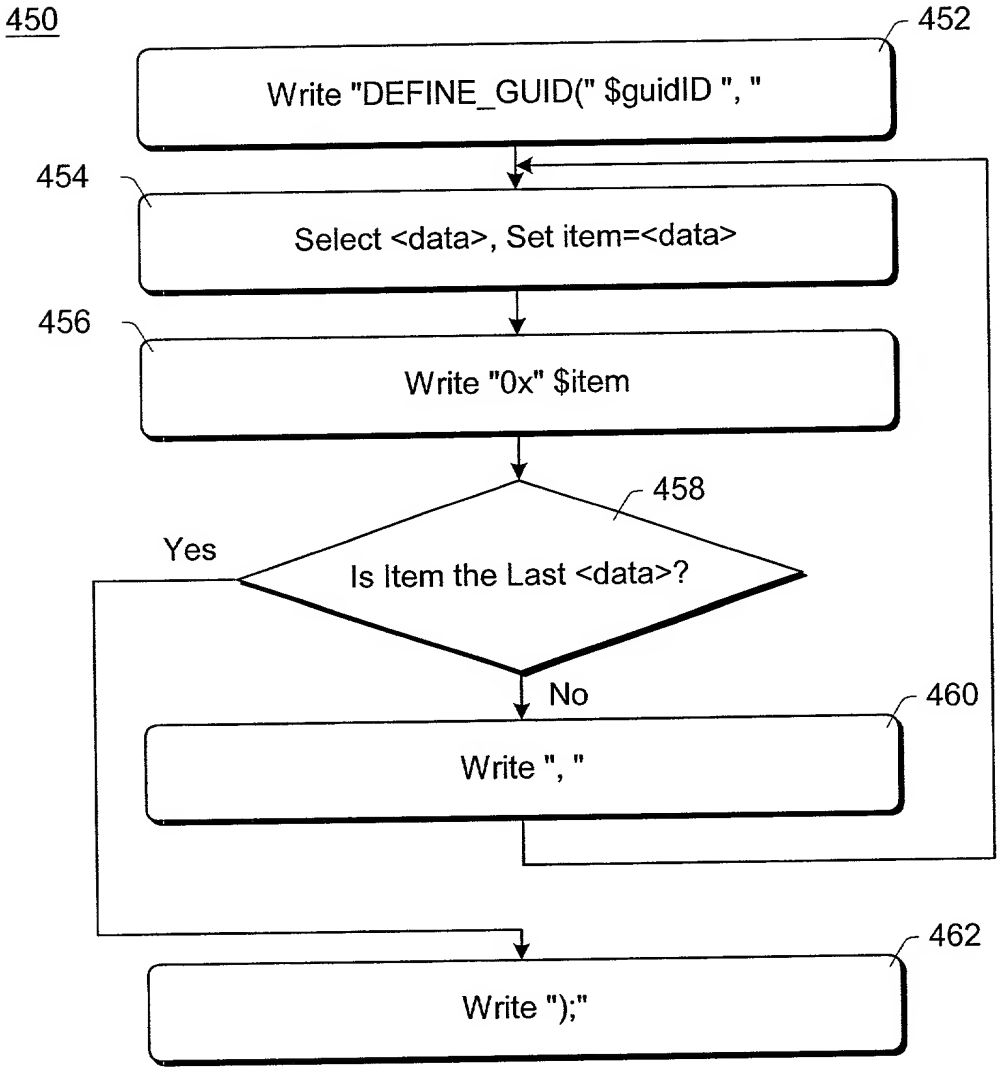

Fig. 13

4807669.02

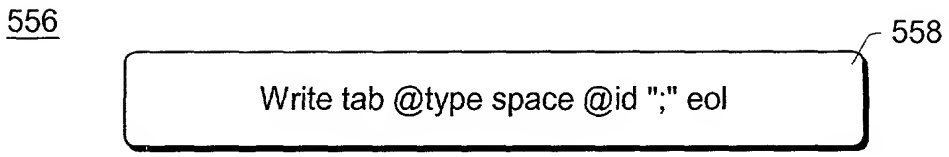

Fig. 14

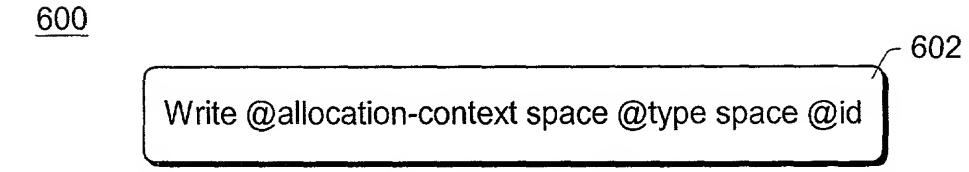

Fig. 15

Microsoft Corporation - 2004

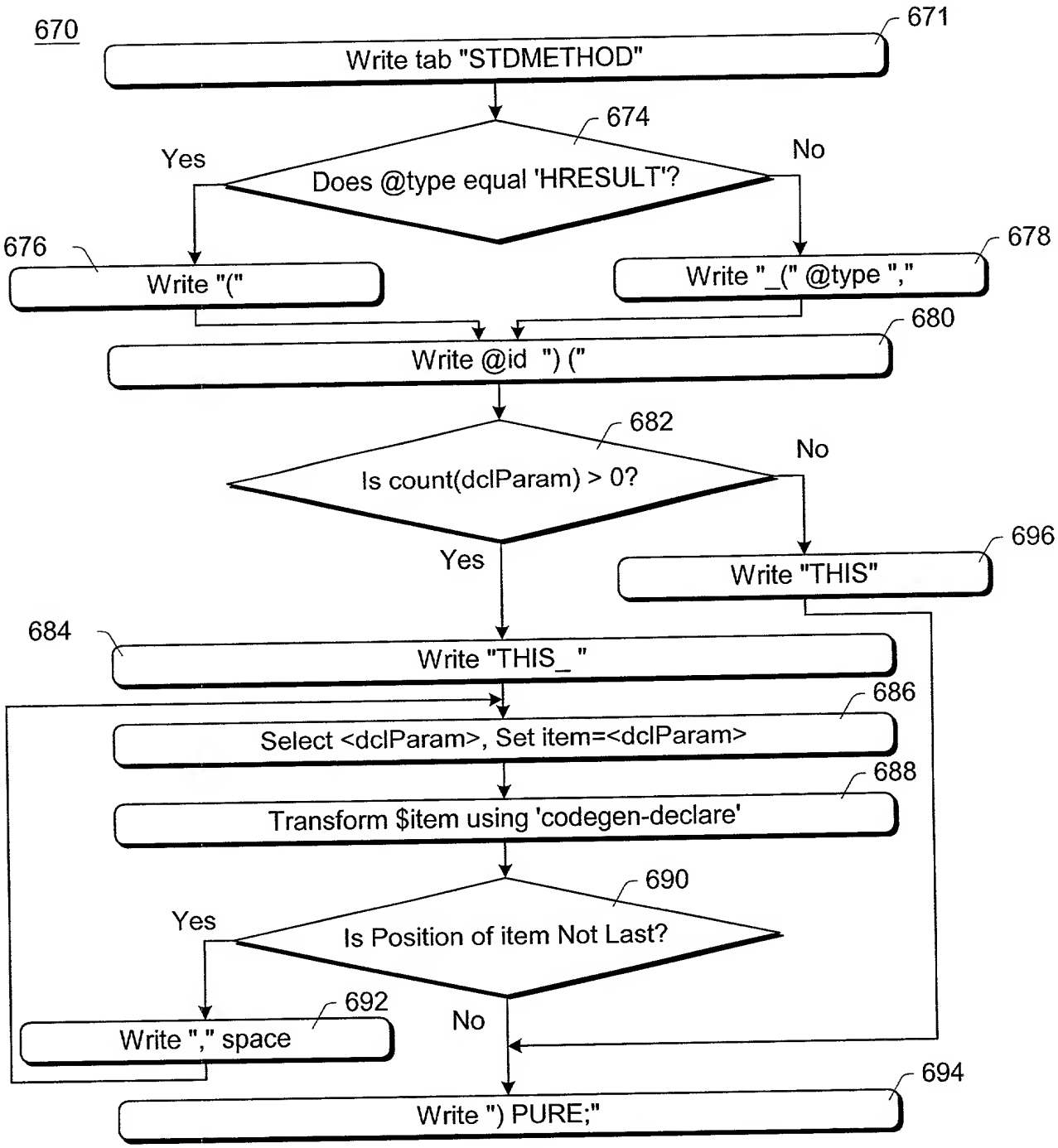
670
671
Write tab "STDMETHOD"
674
Does @type equal 'HRESULT'?
Yes
No
676
Write "("
678
Write "_(" @type ","
680
Write @id ") ("
682
Is count(dclParam) > 0?
No
Yes
696
Write "THIS"
684
Write "THIS_ "
686
Select <dclParam>, Set item=<dclParam>
688
Transform $item using 'codegen-declare'
690
Is Position of item Not Last?
Yes
No
692
Write "," space
694
Write ") PURE;"

Fig. 16

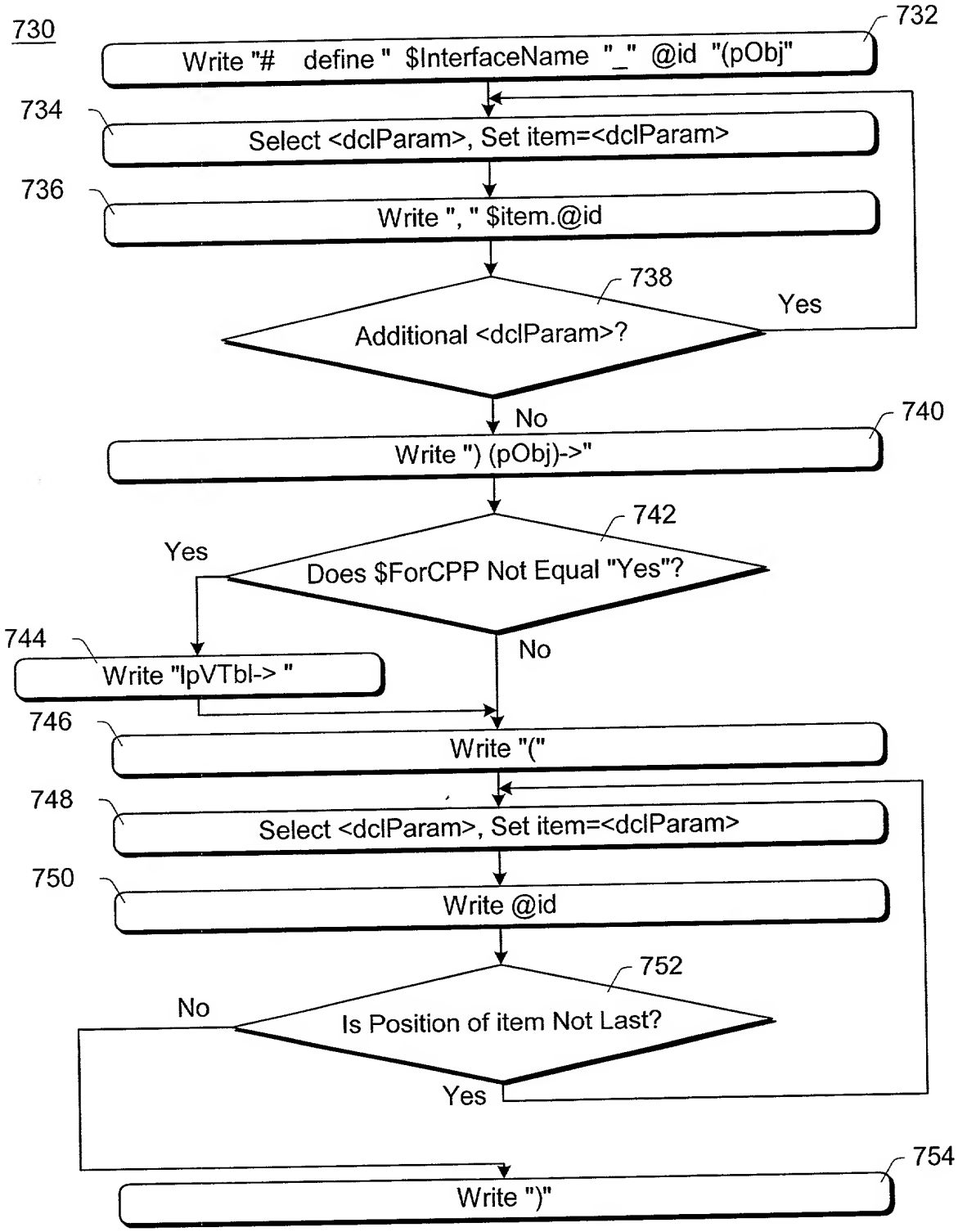
730
732
Write "# define " $InterfaceName "_" @id "(pObj"
734
Select <dclParam>, Set item=<dclParam>
736
Write ", " $item.@id
738
Additional <dclParam>?
Yes
No
740
Write ") (pObj)->"
742
Does $ForCPP Not Equal "Yes"?
Yes
No
744
Write "lpVTbl-> "
746
Write "("
748
Select <dclParam>, Set item=<dclParam>
750
Write @id
752
Is Position of item Not Last?
No
Yes
754
Write ")"

Fig. 17

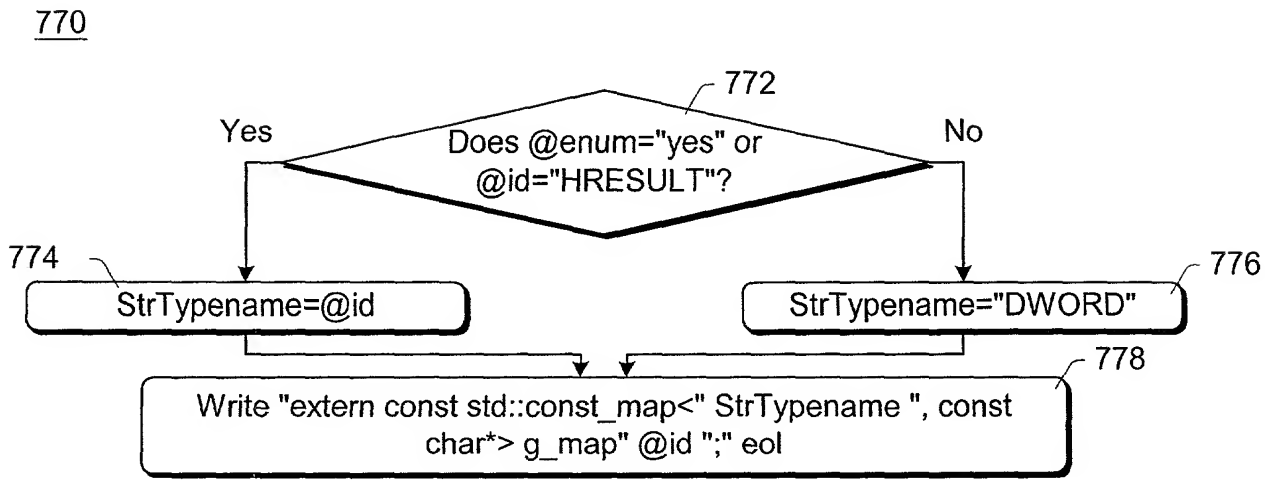

Fig. 18

4007565.02

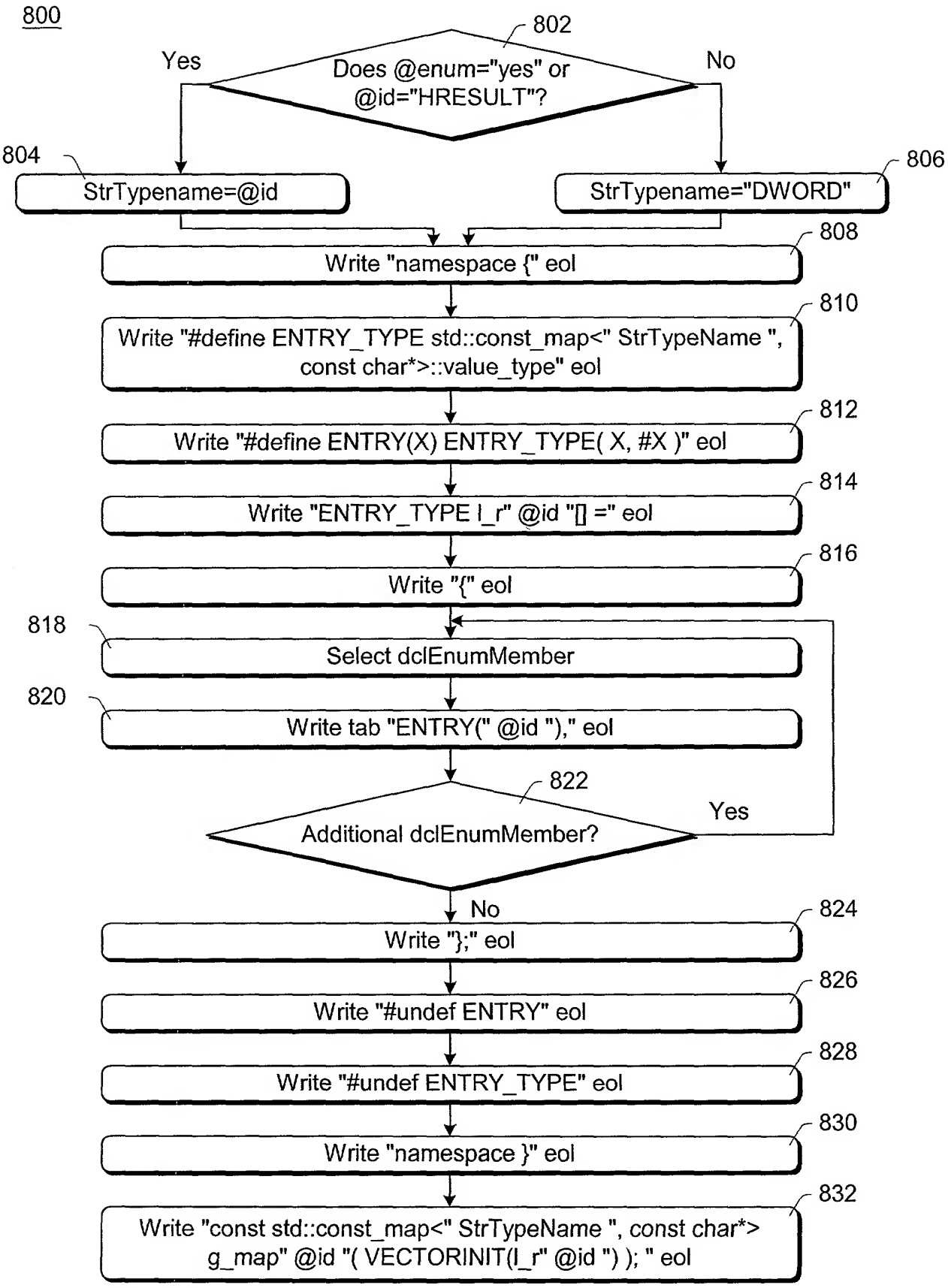
800
802
Does @enum="yes" or @id="HRESULT"?
Yes
No
804
StrTypename=@id
806
StrTypename="DWORD"
808
Write "namespace {" eol
810
Write "#define ENTRY_TYPE std::const_map<" StrTypeName ", const char*>::value_type" eol
812
Write "#define ENTRY(X) ENTRY_TYPE(X, #X)" eol
814
Write "ENTRY_TYPE l_r" @id "[] =" eol
816
Write "{" eol
818
Select dclEnumMember
820
Write tab "ENTRY(" @id ")," eol
822
Additional dclEnumMember?
Yes
No
824
Write "};" eol
826
Write "#undef ENTRY" eol
828
Write "#undef ENTRY_TYPE" eol
830
Write "namespace }" eol
832
Write "const std::const_map<" StrTypeName ", const char*> g_map" @id "(VECTORINIT(l_r" @id ")); " eol

Fig. 19

860

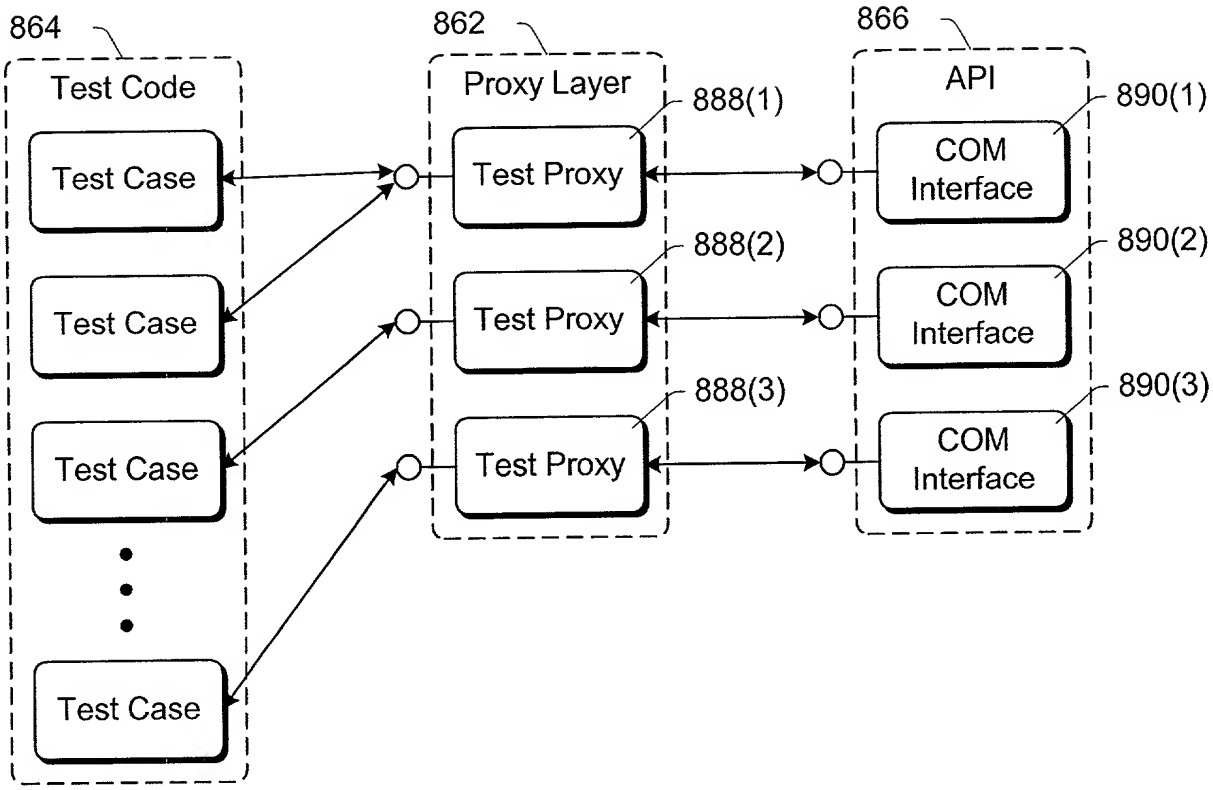

Fig. 20

2934.0267.US1

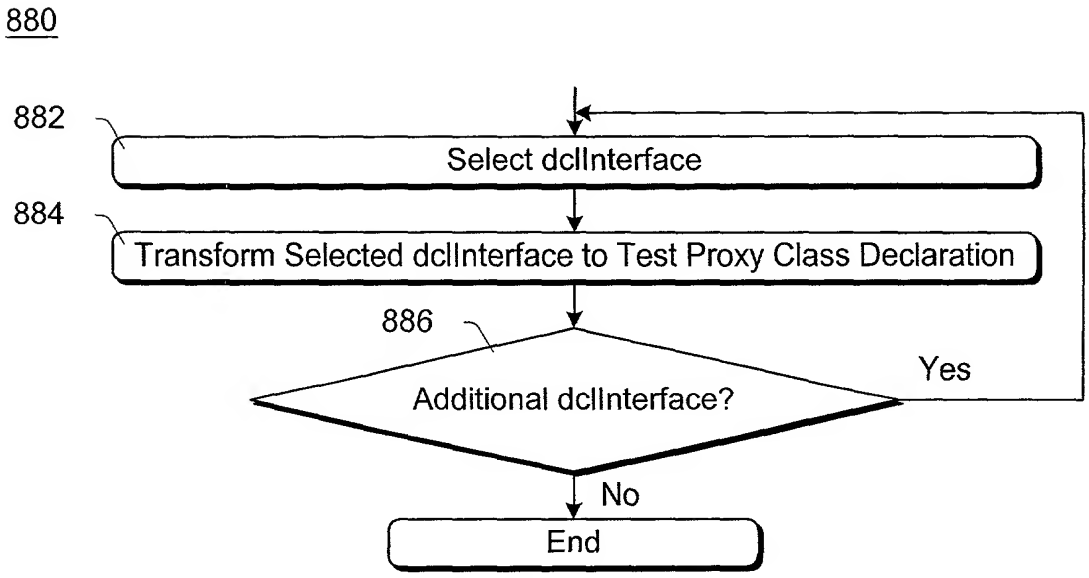

Fig. 21

10076567.062402

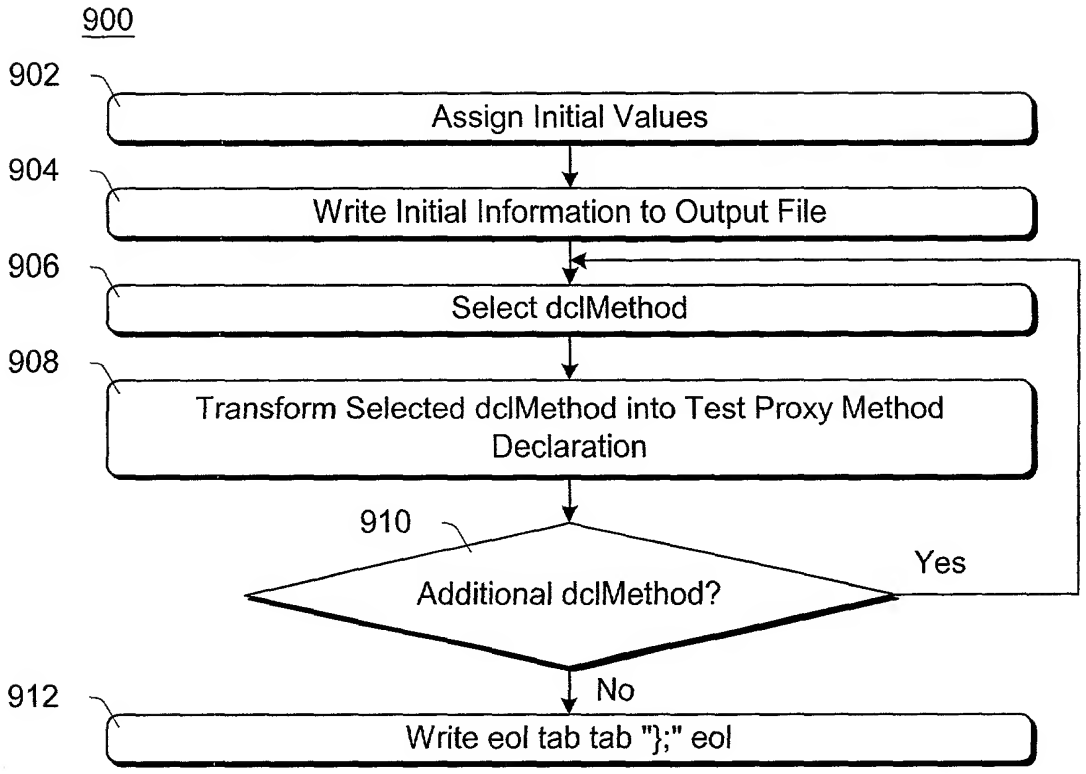

Fig. 22

1005659.02 PA3465

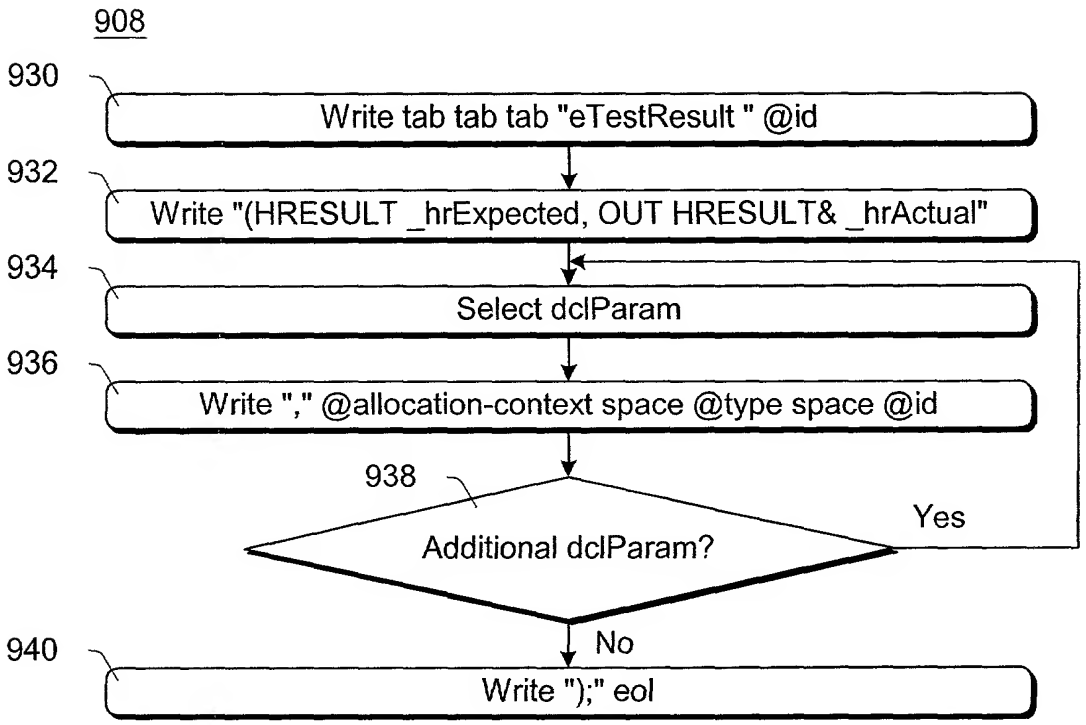

Fig. 23

4006769v.1 0925.4006

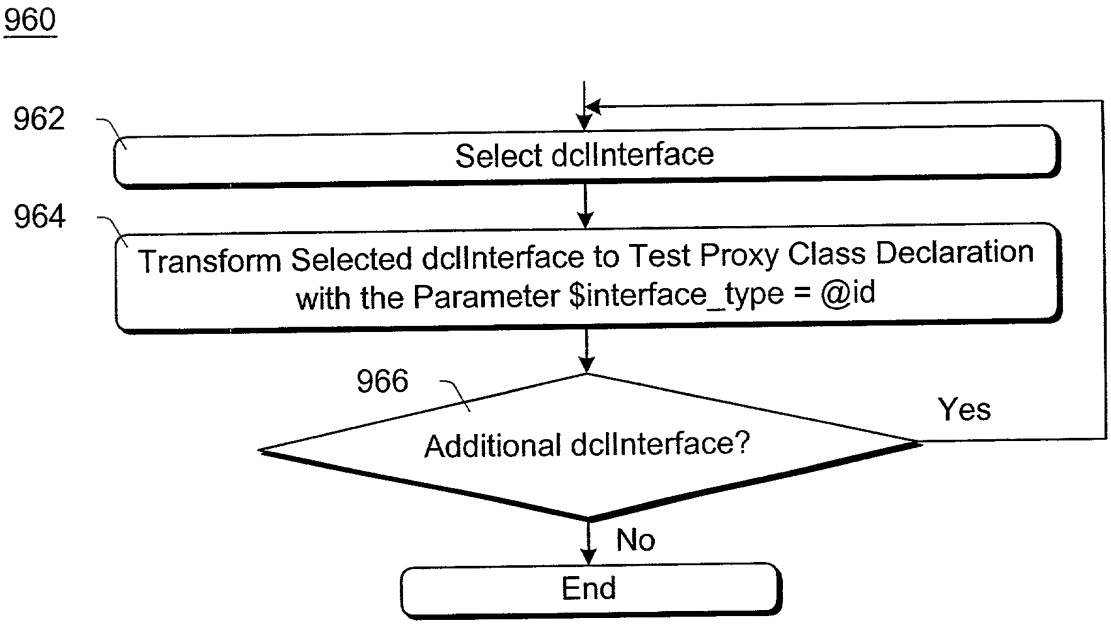

Fig. 24

EC0760 .0024.00

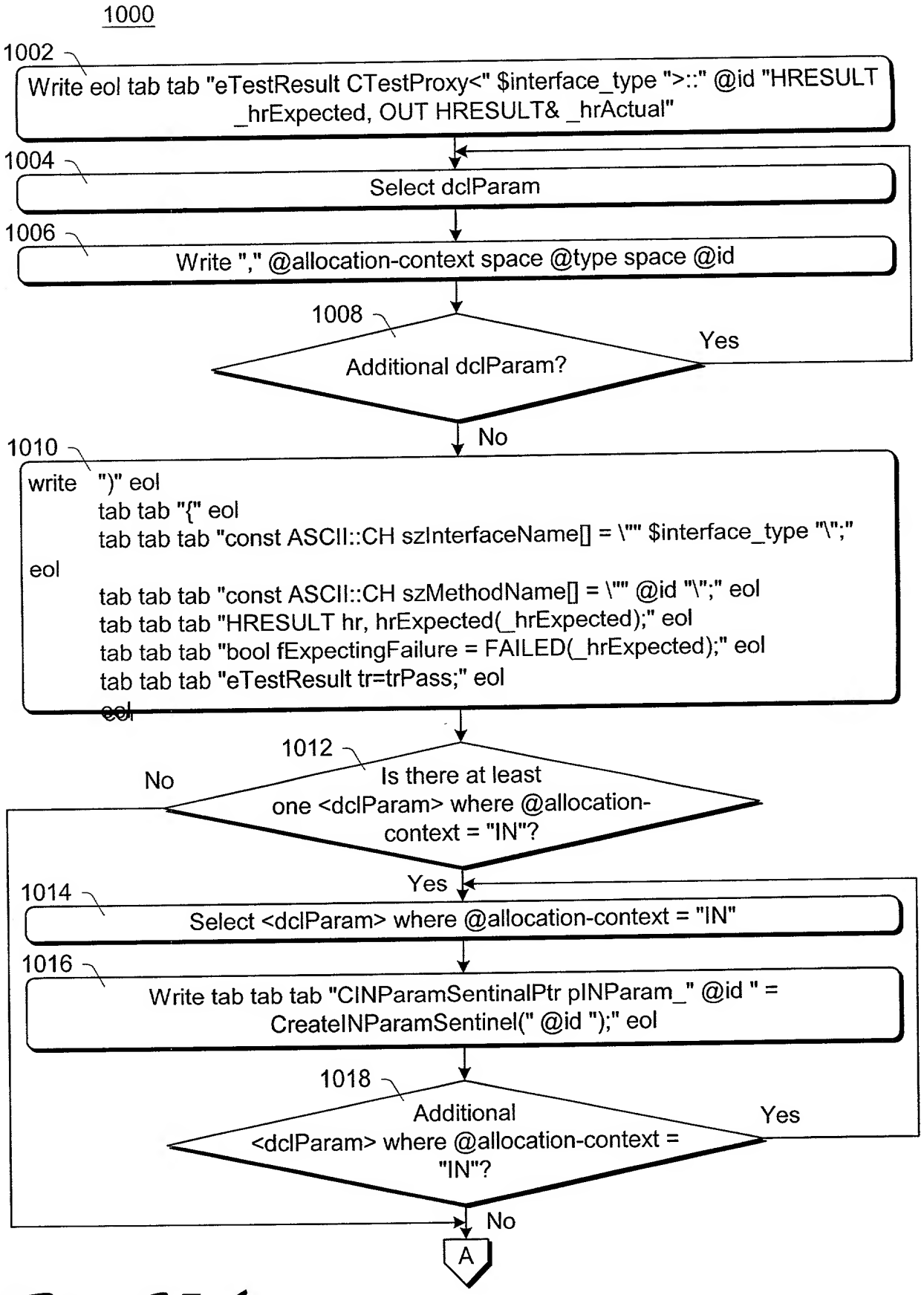

Fig. 25A

4005669.02

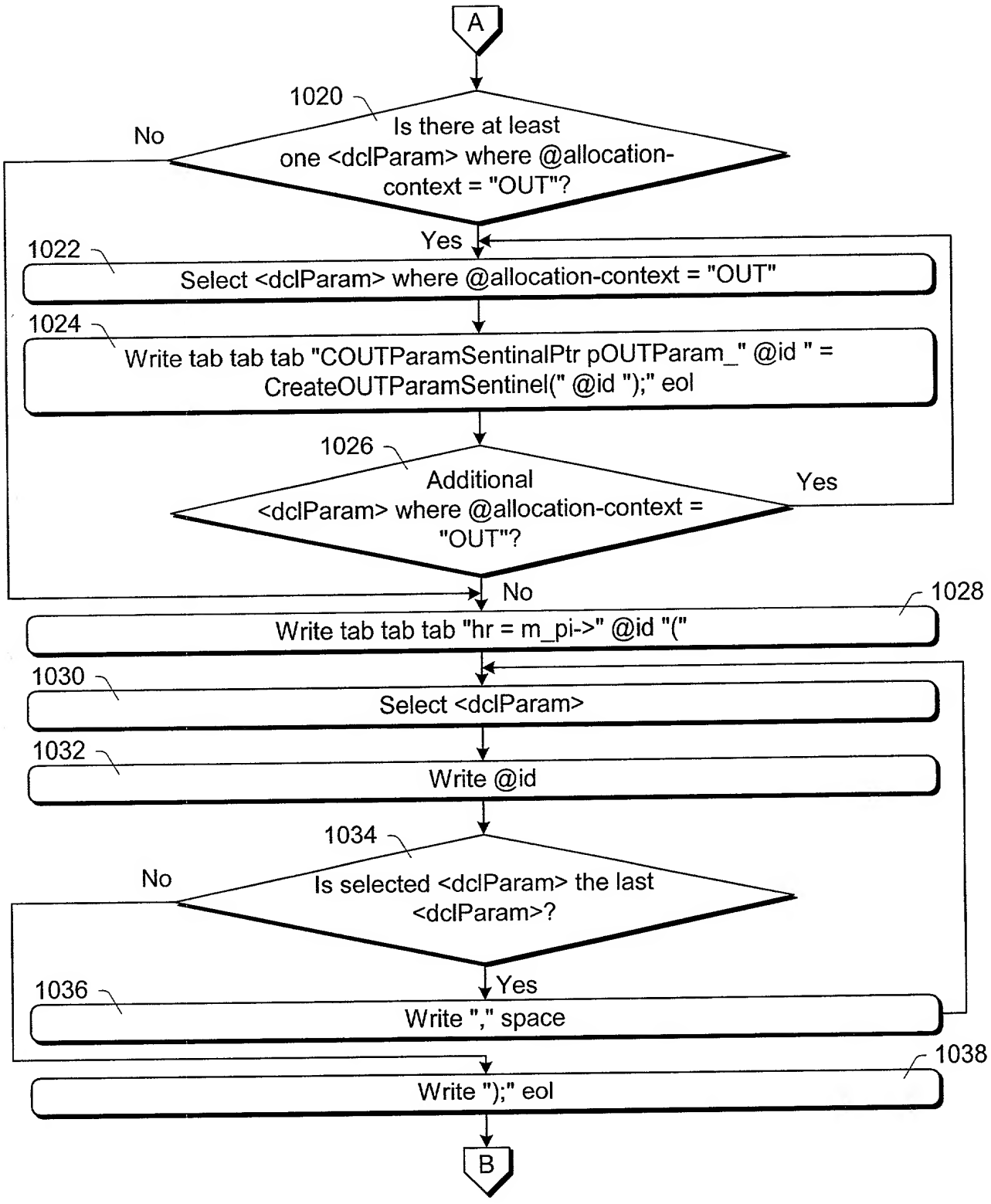
A
1020
Is there at least one <dclParam> where @allocation-context = "OUT"?
No
Yes
1022
Select <dclParam> where @allocation-context = "OUT"
1024
Write tab tab tab "COUTParamSentinalPtr pOUTParam_" @id " = CreateOUTParamSentinel(" @id ");" eol
1026
Additional <dclParam> where @allocation-context = "OUT"?
Yes
No
1028
Write tab tab tab "hr = m_pi->" @id "("
1030
Select <dclParam>
1032
Write @id
1034
Is selected <dclParam> the last <dclParam>?
No
Yes
1036
Write "," space
1038
Write ");" eol
B

Fig. 25B

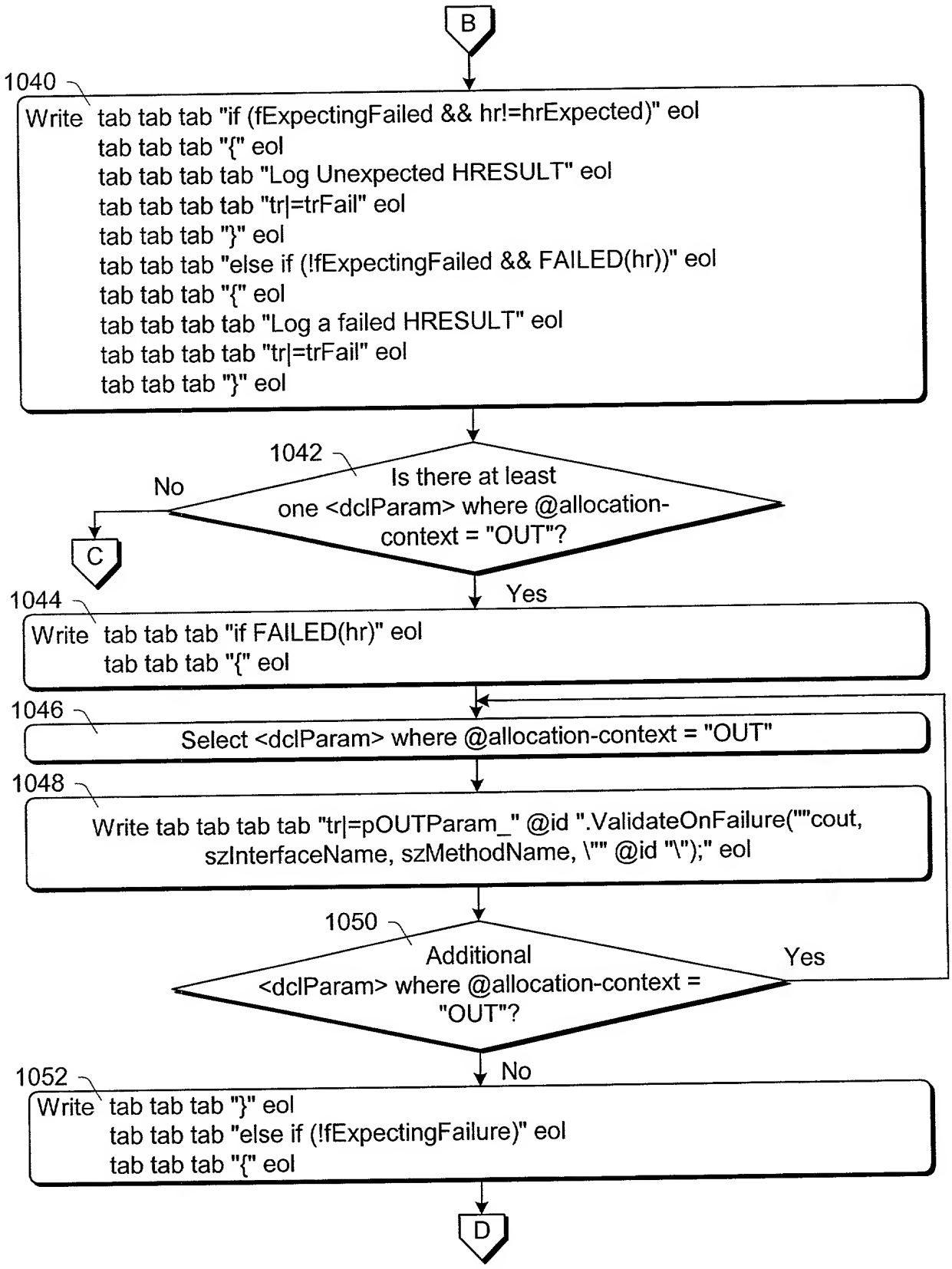

Fig. 25C

1007663v1 - 03450

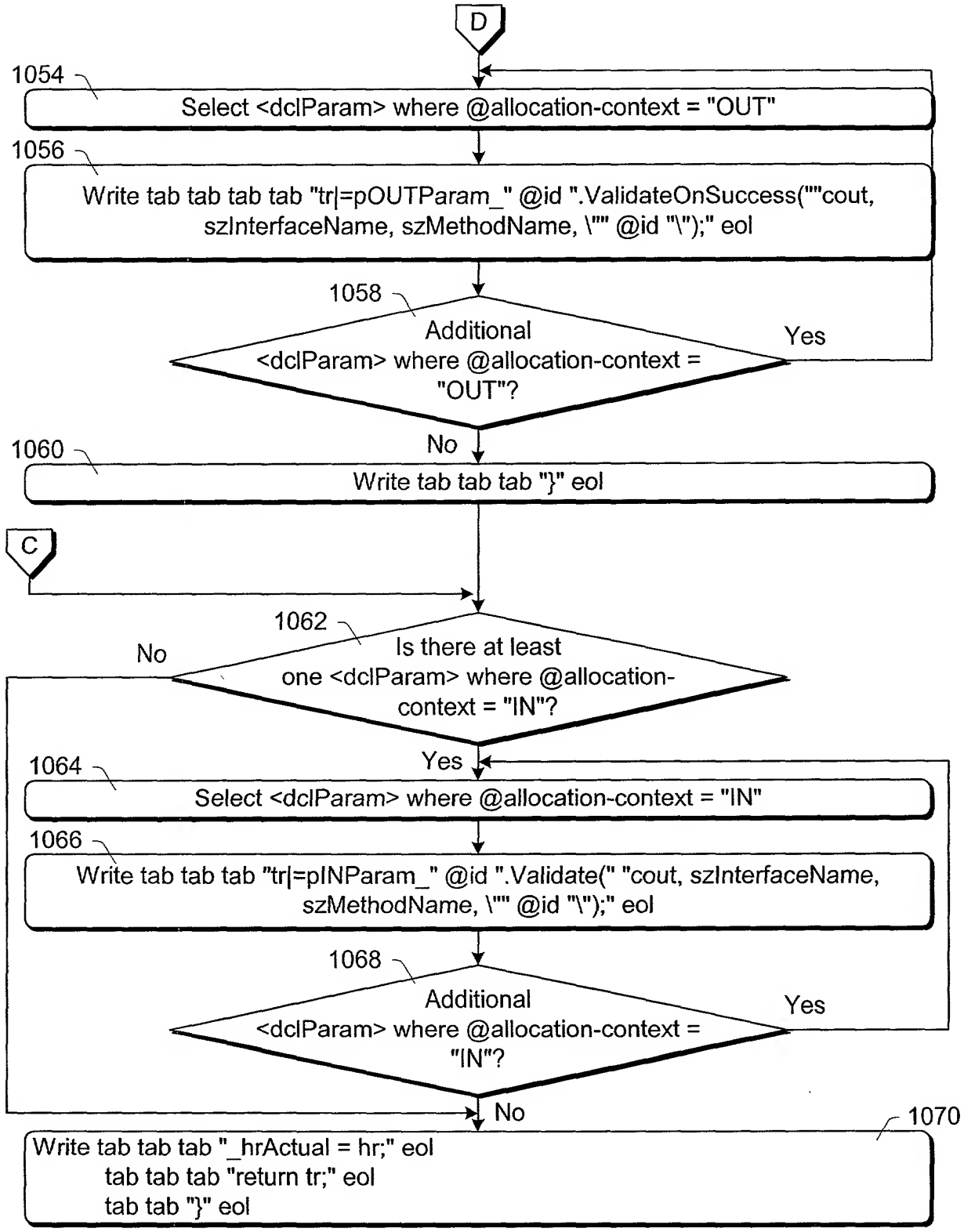
D
1054
Select <dclParam> where @allocation-context = "OUT"
1056
Write tab tab tab tab "tr|=pOUTParam_" @id ".ValidateOnSuccess(""cout, szInterfaceName, szMethodName, \"" @id "\");" eol
1058
Additional <dclParam> where @allocation-context = "OUT"?
Yes
No
1060
Write tab tab tab "}" eol
C
1062
Is there at least one <dclParam> where @allocation-context = "IN"?
No
Yes
1064
Select <dclParam> where @allocation-context = "IN"
1066
Write tab tab tab "tr|=pINParam_" @id ".Validate(" "cout, szInterfaceName, szMethodName, \"" @id "\");" eol
1068
Additional <dclParam> where @allocation-context = "IN"?
Yes
No
1070
Write tab tab tab "_hrActual = hr;" eol
tab tab tab "return tr;" eol
tab tab "}" eol

Fig. 25D

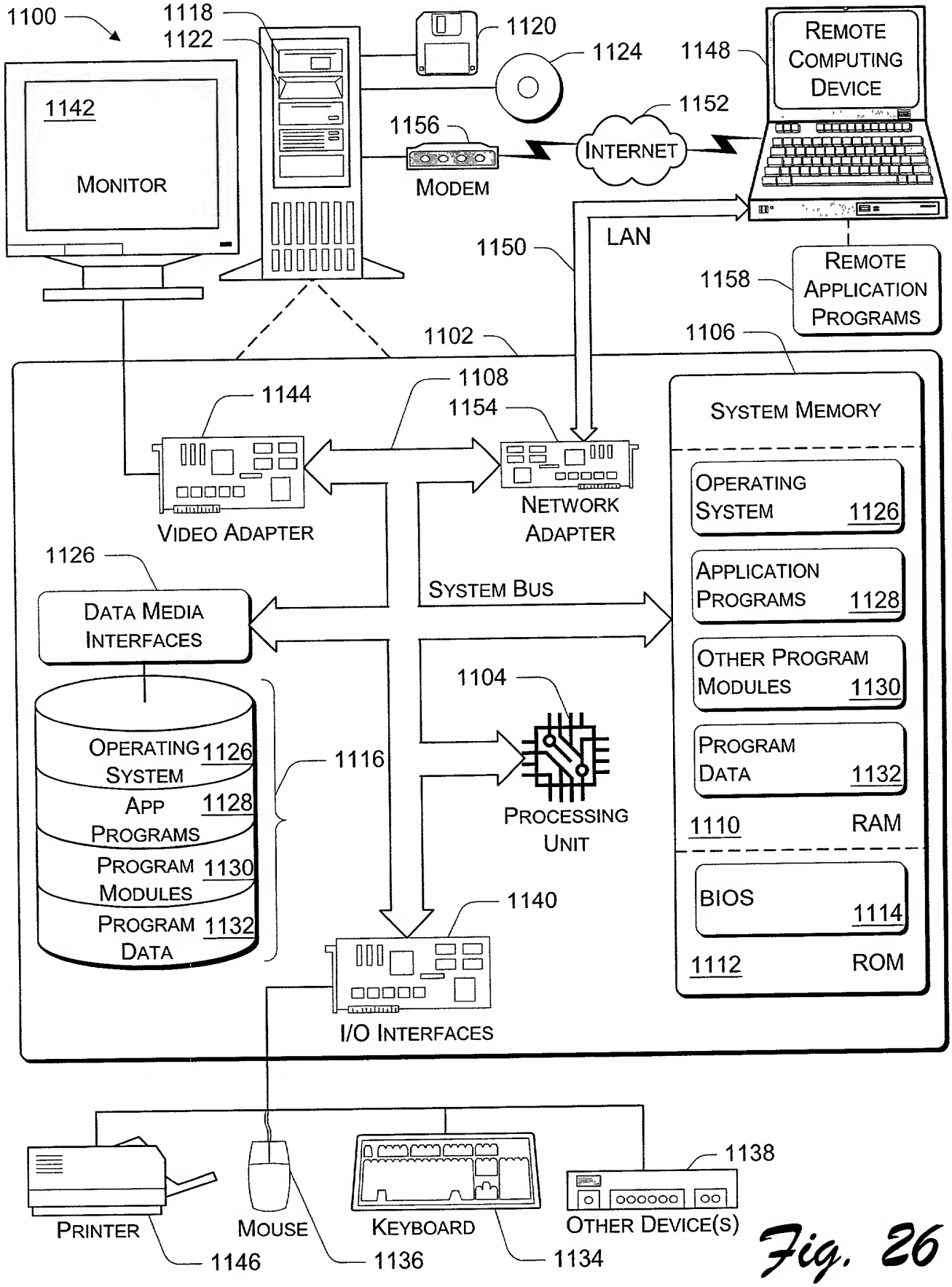

Fig. 26

Doc 5667 02 4502